Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A method for processing a markup language file having one or more portions, the method comprising steps performed by a processor of:

automatically downloading by said processor a first markup language file using the hyper text transfer protocol and referencing by said processor the first markup language file by its uniform resource location (URL) or by a name of a local file on a system on which a user is operating, said first markup language file including arbitrarily named tags;

automatically determining by said processor when the step of downloading is complete and, upon a determination that the step of downloading is complete, automatically parsing by said processor the first markup language file for one or more portions of the first markup language file; and

automatically determining by said processor when the step of parsing is complete and, upon a determination that the step of parsing is complete, automatically storing by said processor each portion of the first markup language file into a directory structure containing folders, subfolders, and files, complying with the structure of the first markup language file, wherein each of the folders and subfolders depend from the tag names in the markup language file.

2. (previously presented) The method of claim 1, wherein if there are any relative uniform resource locations, automatically converting said relative uniform resource locations into absolute uniform resource locations.

3. (currently amended) The method of claim 1, comprising providing a command language set allowing selection, viewing and other processing of the one or more ~~objects~~ portions of the first markup language file, the command language set comprising a plurality of commands for selection.

4. (original) The method of claim 3, wherein said command language set comprises a command for listing the contents of a folder.

5. (original) The method of claim 3, wherein said command language set comprises a command for changing folders and syntax for designating subfolders of folders.

6. (original) The method of claim 3, wherein said command language set comprises a command for listing the contents of a file.

7. (original) The method of claim 3, wherein said command language set comprises a command for listing the attributes of a hyper text markup language tag.

8. (original) The method of claim 3, wherein said command language set comprises a command for using wildcards in any pathname.

9. (original) The method of claim 3, wherein said command language set comprises a method for treating the contents of a file as a local variable when a directory pathname for the files is referenced.

10. (original) The method of claim 3, wherein said command language set comprises a method for treating an attribute of a folder or file as a local variable when a directory pathname is referenced.

11. (original) The method of claim 3, wherein said command language set comprises a command for making new folders in the directory structure.

12. (original) The method of claim 3, wherein said command language set comprises a command for making new files in the directory structure.

13. (original) The method of claim 3, wherein said command language set comprises a command for copying folders in the directory structure.

14. (original) The method of claim 3, wherein said command language set comprises a command for recursively copying folders in the directory structure.

15. (original) The method of claim 3, wherein said command language set comprises a command for copying files in the directory structure.

16. (original) The method of claim 3, wherein said command language set comprises a command for recursively copying files in the directory structure.

17. (original) The method of claim 3, wherein said command language set comprises a command for renaming folders in the directory structure.

18. (original) The method of claim 3, wherein said command language set comprises a command for renaming files in the directory structure.

19. (original) The method of claim 3, wherein said command language set comprises a command for creating new files through redirecction of an output of a command.

20. (original) The method of claim 3, wherein said command language set comprises a command for setting a file value.

21. (original) The method of claim 3, wherein said command language set comprises a command for saving the modified first markup language file to a disk.

22. (original) The method of claim 3, wherein said command language set comprises a command for outputting the modified first markup language file to standard output.

23. (original) The method of claim 3, wherein said command language set comprises allowing creation of one or more executable batch files containing a subset of the set of commands.

24. (original) The method of claim 23, comprising defining local variables for processing in conjunction with variables and attributes of the files.

25. (original) The method of claim 24, comprising a command for creating loop processing.

26. (original) The method of claim 24, comprising a command for jumping to a new specified location within the batch file and resuming execution at the new location.

27. (previously presented) The method of claim 3, comprising creating XML formatted documents that contain references to tag variables, allowing insertion into a markup language document the contents of a folder, subfolder or file from a second markup language document.

28. (currently amended) A system for processing a markup language file having one or more portions, the system having a computer having at least ~~a memory,~~ a processor, a memory operably coupled to said memory, ~~and a computer-readable medium having software code embedded thereon, which, when said code is loaded into said memory and run by said processor, causes said processor to perform steps of~~ said memory being configured for storing a computer program executable by said processor, said computer program comprising:

computer program code for automatically downloading by said processor a first markup language file using the hyper text transfer protocol and automatically referencing by said processor the first markup language file by its uniform resource location (URL) or by a name of a local file on a system on which a user is operating, said first markup language file including arbitrarily named tags;

computer program code for automatically determining by said processor when the steps of downloading and referencing are complete and, upon a determination that the steps of downloading and referencing are complete, computer program code for automatically parsing by said processor the first markup language file for one or more portions of the first markup language file; and

computer program code for automatically determining by said processor when the step of parsing is complete and, upon a determination that the step of parsing is complete, computer program code for automatically storing by said processor each portion of the first markup language file into a directory structure containing folders, subfolders, and files, complying with the structure of the first markup language file, wherein each of the folders and subfolders depend from the tag names in the markup language file.

29. (new) The method of claim 1 wherein said URL is read from a browser via a CGI transaction.

30. (new) The method of claim 1 wherein said URL is read from a database.

31. (new) The method of claim 1 wherein said URL is embedded in computer program code.

32. (new) The method of claim 1 wherein said URL is in an environment variable.

33. (new) The method of claim 1 wherein said URL is in a local variable.

34. (new) The method of claim 1 wherein said URL is in a global variable.

35. (new) The system of claim 28 wherein said computer program further comprises computer program code for reading said URL from a browser via a CGI transaction.

36. (new) The system of claim 28 wherein said computer program further comprises computer program code for reading said URL from a database.

37. (new) The system of claim 28 wherein said computer program further comprises computer program code for reading said URL embedded in computer program code.

38. (new) The system of claim 28 wherein said computer program further comprises computer program code for reading said URL in an environment variable.

39. (new) The system of claim 28 wherein said URL is in a local variable.

40. (new) The system of claim 28 wherein said URL is in a global variable.